INTERACTIVE ONLINE TUTORIAL SYSTEM WITH USER ASSISTANCE FUNCTION FOR SOFTWARE PRODUCTS

This is a continuation of application Ser. No. 07/858,378, filed Mar. 26, 1992, now abandoned.

BACKGROUND OF THE INVENTION

The present invention relates generally to tutorial systems for providing instruction in the operation of software products. More particularly, the invention pertains to online tutorial systems wherein software product training is provided interactively by guiding the user through one or more examples of product operation. Still more particularly, the invention involves a computer implemented, interactive tutorial system providing online instruction, monitoring and assistance in conjunction with a contemporaneously operating software product.

As software developers strive to provide ever more versatile and powerful products, the task of becoming proficient in the use and implementation of such products becomes increasingly difficult. Even for those users who consider themselves computer "literate", days and sometimes weeks of instruction may be required before the user is able to take full advantage of functional capabilities offered by many software products. In such cases, the quality of the instructional materials accompanying the software may be nearly as important as the product in ensuring ultimate user satisfaction.

Software instructional materials for prior art products fall into two general types. On one hand are product documentation materials, such as printed manuals and online product description files. These materials provide textual descriptions of operational features of the product. They may also include examples of product operations. Although documentation of this nature may be adequate for reference purposes, such materials are not particularly suitable for providing initial instruction in the use of a product. Most users would rather use the product directly than spend hours reading ancillary instructional materials. Accordingly, a number of products follow a learn-by-doing approach using interactive tutorial materials, which represent a second category of instructional materials.

Unlike product documentation materials, tutorials provide instruction in a manner which gives the user an opportunity to gain practical experience in the context of solving actual problems. Both online and workbook-based tutorials have been proposed wherein the user is guided through a presentation of one or more examples or problems in a sequence of lessons. In workbook-based systems the student performs specified actions on the product in accordance with a lesson plan set forth in a printed workbook. Online systems work in a similar fashion except that the tutorial lessons are displayed sequentially on-screen. In addition, online tutorials typically include the capability of monitoring student actions and advising when a mistake has been made. Moreover, if the user requires assistance, a preprogrammed demonstration can be requested to perform the correct action(s) to be taken. For these reasons, online tutorials represent the most promising avenue for providing instruction in the operation of software products.

Most prior art online tutorial systems allow users to interact with a simulation of the software product of interest. This approach to tutorial instruction is both inefficient and costly. The software designer must not only code and debug the application product itself but also the duplicative simulation code for the tutorial. The simulation code must be periodically updated and maintained as the product changes, all at additional time and cost. The effectiveness of simulation tutorials as a teaching vehicle is also open to question. Simulation tutorials provide only limited product interaction because the simulations can only emulate the user interface of the product and do not have full capabilities in processing user actions. The user is not afforded direct experience with the product and thus may be deprived of a full and accurate understanding of product characteristics. Because only a partial view of the product is provided, users are unable to explore other portions of the product or learn on their own beyond what the tutorial covers.

As a solution to the inefficiencies of simulation tutorials, several systems have been proposed wherein online tutorial instruction proceeds in conjunction with the operation of the product to be taught. The prior art includes interactive tutorials implemented in single task operating systems in which special hardware or software subsystems are required to control the environment and provide tutorial information. This special hardware and software includes such things as interrupts between the user and operating system, processing outside the operating system, and external information presentation hardware. As a result, the flexibility and efficiency of these tutorial products is relatively limited. For example, the information presentation function in many prior art tutorial systems is provided in a strict top down sequence which must be followed in the order set forth. No provision is made for selectively controlling the level of tutorial information detail presented as the student works through the lesson. In other cases, the input monitoring function of prior art tutorial systems is unduly restrictive insofar as literal input response from the student is required. No provision is made for allowing the student to provide input to the computer for performing tasks which are non critical to the product. Finally, in other prior art tutorial systems wherein a demonstration mode is provided, no provision is made for selectively controlling the duration of the automated control sequence. Typically, the demonstration sequence encompasses a predetermined number of input steps.

The foregoing disadvantages of prior art tutorial systems stem in part from limitations inherent in single task operating systems and in user interface software that provides only limited control functionality. Recent advances in software technology offer opportunities to create tutorial products with better user interfaces and more sophisticated program control without undue product complexity. The OS/2 TM operating system from IBM Corporation provides a multi-tasking environment wherein multiple products can be run concurrently within the operating system. The OS/2 TM operating system includes a set of programming resources referred to as the application program interface (API). The API in turn provides a subset of support routines collectively referred to as the Presentation Manager TM API, which allows applications to run in rectangular subdivisions of the display called windows. Each windowed application can share the screen with other windowed applications. Each includes a graphics-based user interface to present visual controls for managing, via mouse and keyboard input, the application and files